

EXHIBIT 1

(12) **United States Patent**
Bakshi et al.

(10) **Patent No.:** **US 8,189,566 B2**
(45) **Date of Patent:** **May 29, 2012**

(54) **SOFTWARE BASED TRADING TURRET**

(75) Inventors: **Aseem Bakshi**, Stamford, CT (US);
Rajnish Jain, Fairfield, CT (US);
Andrew G. Klaiber, Trumbull, CT
(US); **Kathleen N. Udall**, Oxford, CT
(US); **Ravi K. Vankayala**, Milford, CT
(US)

(73) Assignee: **IPC Systems, Inc.**, Jersey City, NJ (US)

(*) Notice: Subject to any disclaimer, the term of this
patent is extended or adjusted under 35
U.S.C. 154(b) by 487 days.

(21) Appl. No.: **12/476,490**

(22) Filed: **Jun. 2, 2009**

(65) **Prior Publication Data**

US 2009/0304168 A1 Dec. 10, 2009

Related U.S. Application Data

(60) Provisional application No. 61/059,143, filed on Jun.
5, 2008.

(51) **Int. Cl.**
H04L 12/66 (2006.01)

(52) **U.S. Cl.** **370/352**; 370/463; 379/156

(58) **Field of Classification Search** 370/352,
370/463; 379/156
See application file for complete search history.

(56) **References Cited**

U.S. PATENT DOCUMENTS

6,212,177 B1 * 4/2001 Greene et al. 370/352

* cited by examiner

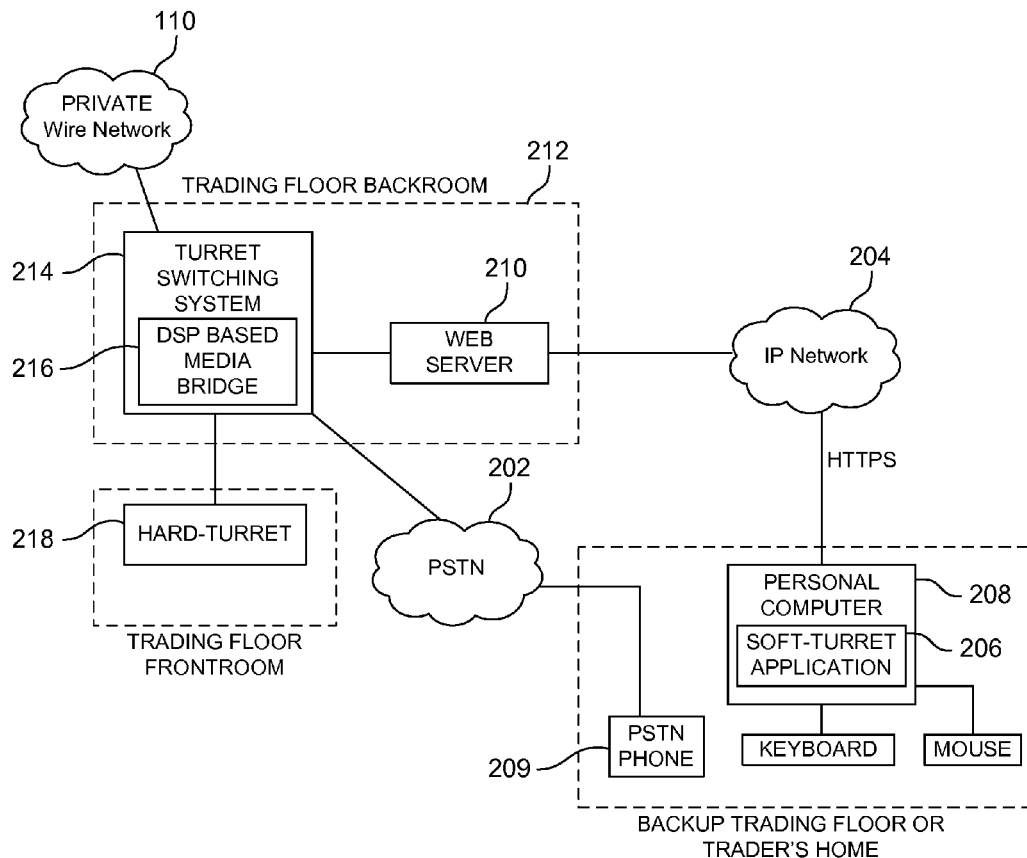
Primary Examiner — Creighton Smith

(74) *Attorney, Agent, or Firm* — Fitzpatrick, Cella, Harper &
Scinto

(57) **ABSTRACT**

A communications system including a turret switching sys-
tem constructed to communicate to a Web server, a turret
device, and to a remote communications device via a first
communications network, the Web server being constructed
to communicate to a client device via a second communica-
tions network, and the client device constructed to control
switching across a plurality of lines.

18 Claims, 9 Drawing Sheets



U.S. Patent

May 29, 2012

Sheet 1 of 9

US 8,189,566 B2

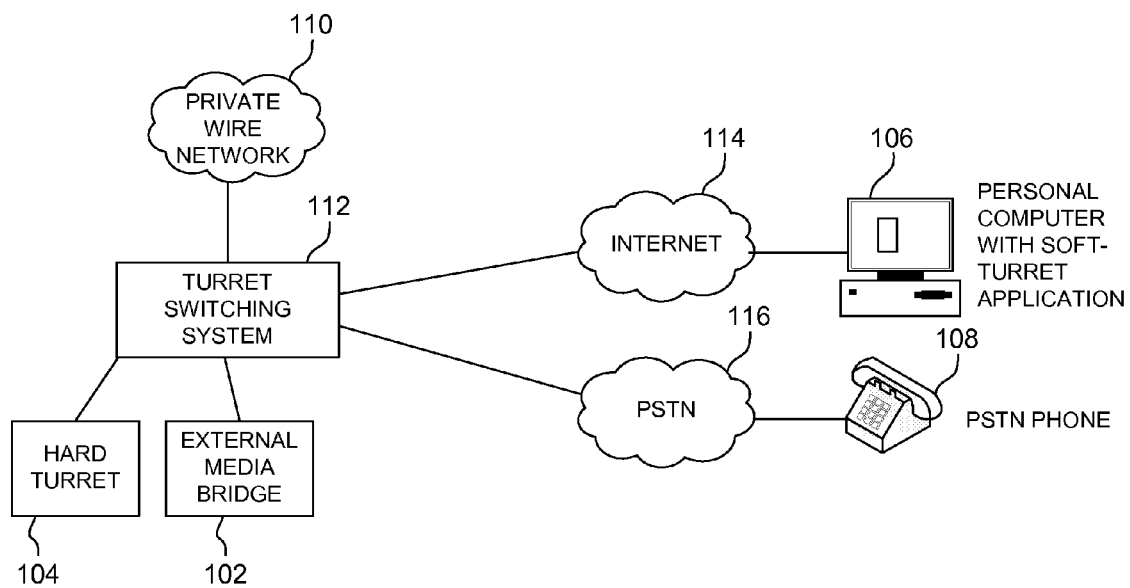


FIG. 1
(PRIOR ART)

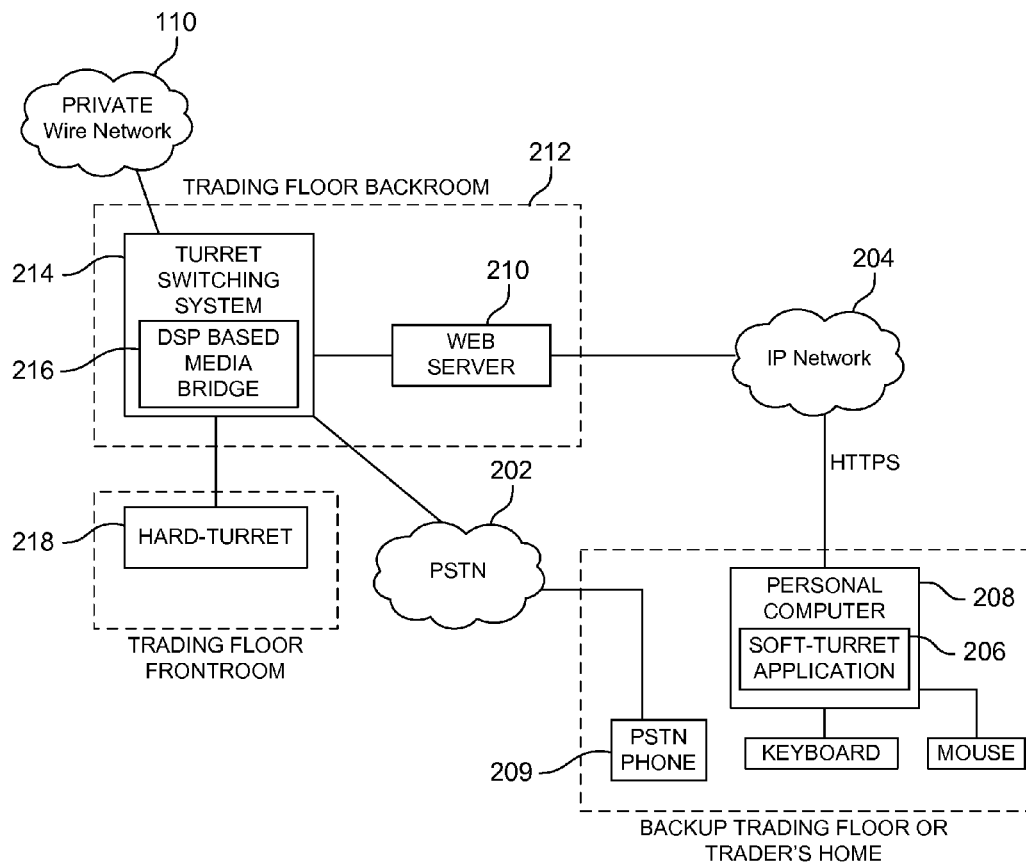


FIG. 2

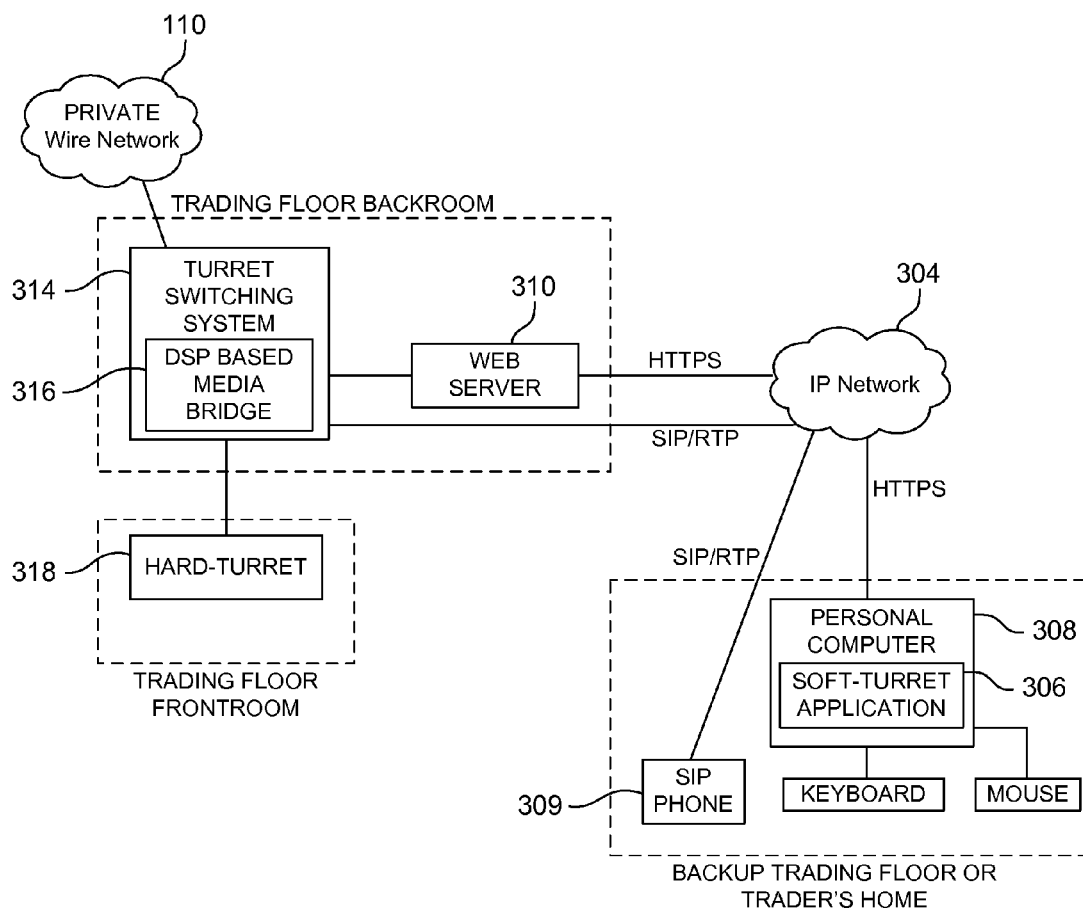


FIG. 3

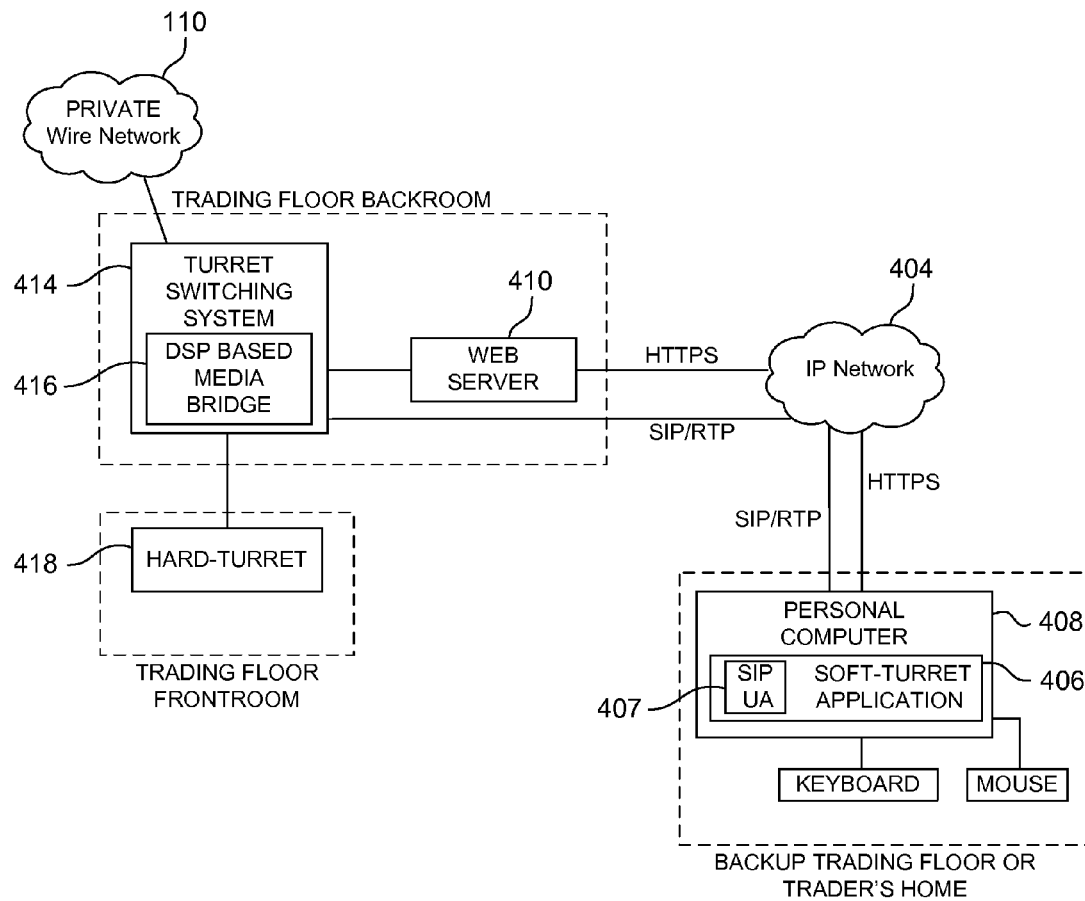
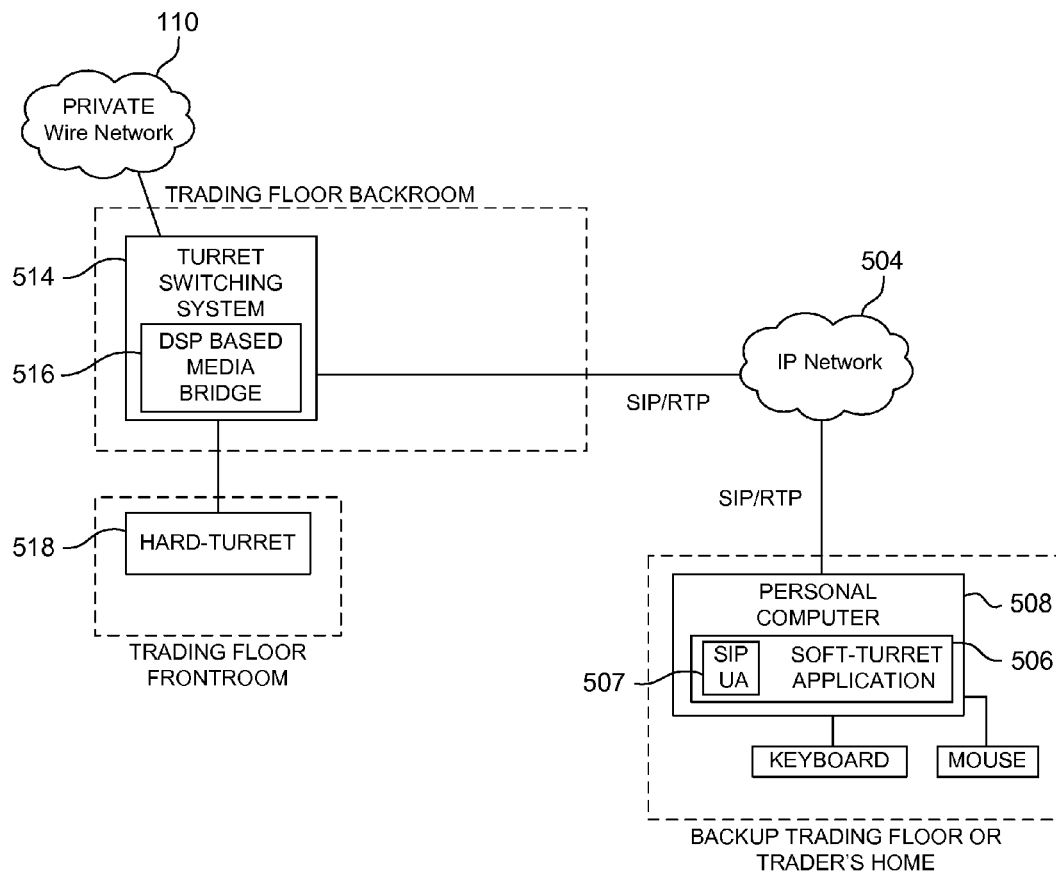


FIG. 4

**FIG. 5**

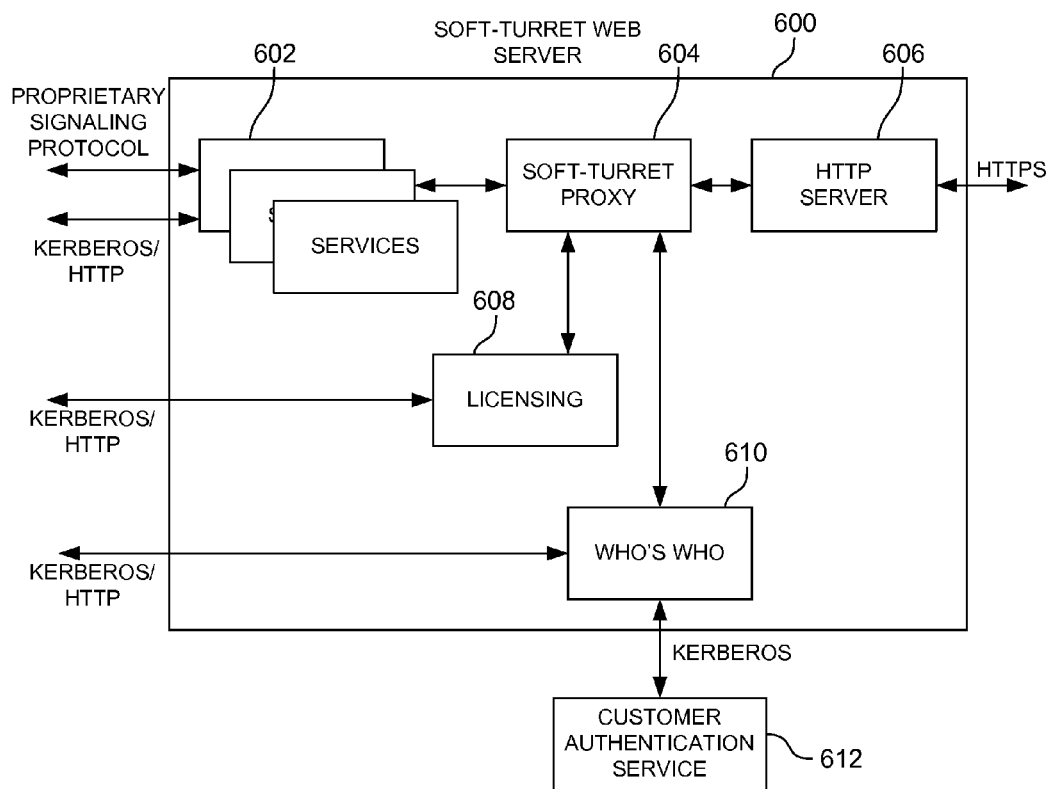
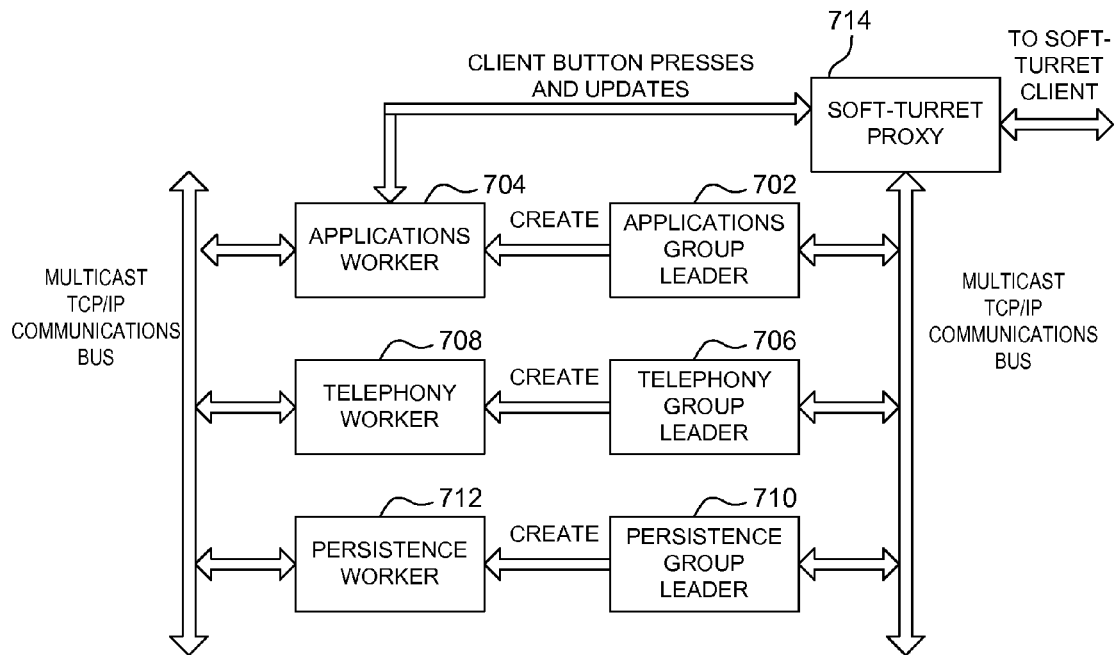


FIG. 6

U.S. Patent

May 29, 2012

Sheet 7 of 9

US 8,189,566 B2**FIG. 7**

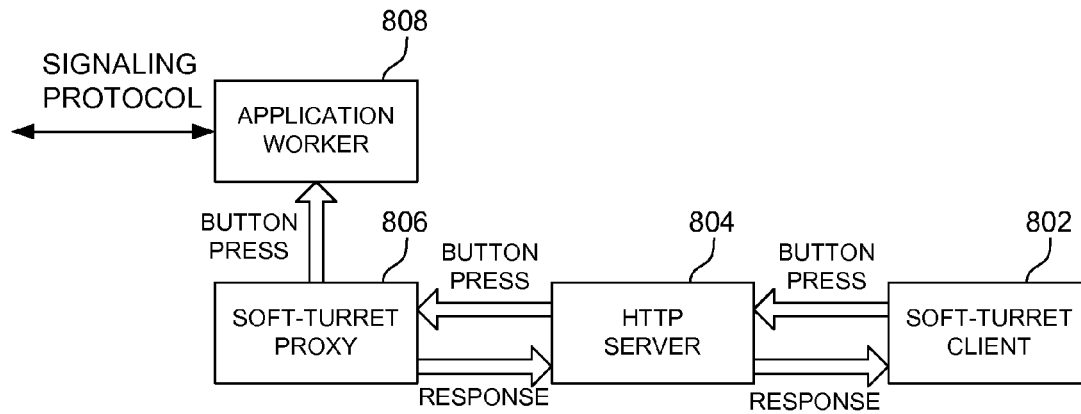


FIG. 8

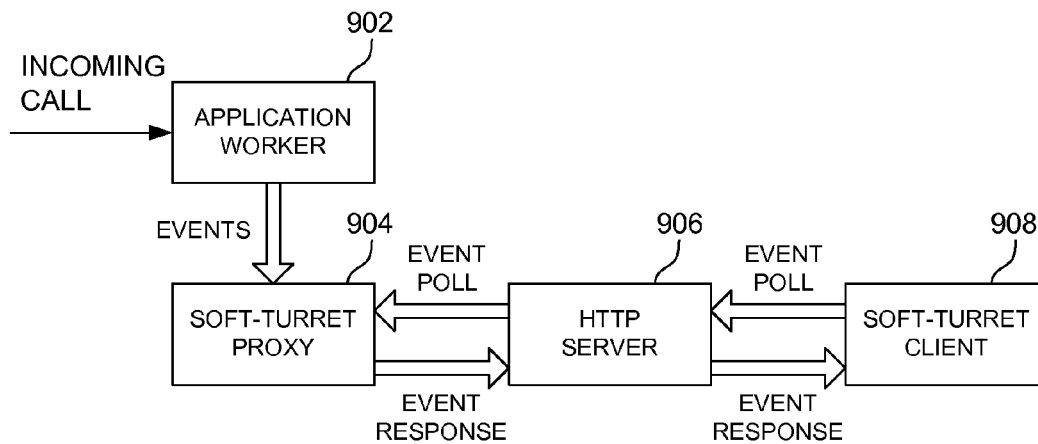
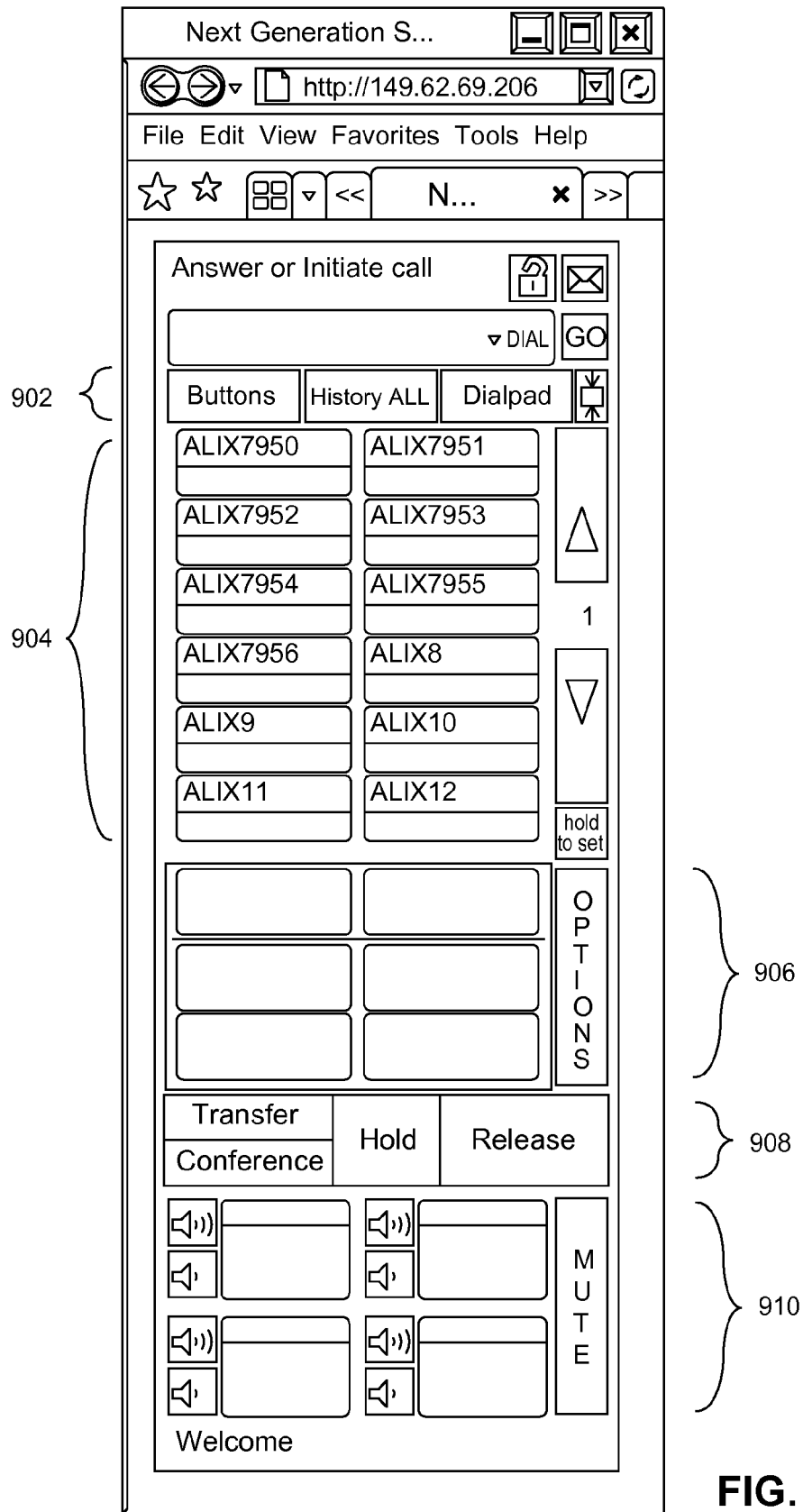


FIG. 9

U.S. Patent**May 29, 2012****Sheet 9 of 9****US 8,189,566 B2****FIG. 10**

US 8,189,566 B2

1

SOFTWARE BASED TRADING TURRET**CROSS REFERENCE TO RELATED APPLICATIONS**

This application claims priority to, and the benefit of, U.S. Provisional Patent Application Ser. No. 61/059,143, filed Jun. 5, 2008, which is hereby incorporated by reference in its entirety.

BACKGROUND OF THE INVENTION**1. Field of the Invention**

The present invention generally relates to telecommunication systems, and more particularly to key telephone systems and trading turrets.

2. Description of the Related Art

A trading turret system is a specialized telephony switching system that allows a relatively small number of users to access a large number of external lines and provides enhanced communication features such as hoot-n-holler, push-to-talk, intercom, video and large-scale conferencing. These features are often utilized in the financial industry such as trading floor environments, as well as security/law enforcement, utilities, healthcare, and customer support (e.g., contact centers) environments.

Users interface with a turret switching system through a trading turret device, which can be implemented in dedicated hardware (sometimes referred to as a “hard” turret) or in a general-purpose computer (sometimes referred to as a “soft” turret). A hard-turret is a phone-like desktop device with multiple handsets, speakers and buttons. A soft-turret is a software application that runs on a trader’s desktop personal computer (PC) or on a mobile computer such as a PDA (Personal Digital Assistant). A soft-turret application may be controlled using the native control interface that the computer provides such as a touch screen, stylus, click wheel, or mouse and keyboard. In addition to rendering a graphical image of the turret on the PC screen, the soft-turret application may also provide voice, instant messaging and presence features.

With the introduction of Session Initiation Protocol (“SIP”) based architectures, new features and interoperability with a wide variety of SIP enabled devices is possible. SIP is an application-layer control (i.e., signaling) protocol for creating, modifying, and terminating sessions such as Internet telephony calls with one or more participants and is defined in RFC-3261, “SIP: Session Initiation Protocol”. SIP has been used in typical IP based networks as the predominant way of signaling between stations and telephony systems and as the trunking protocol between telephony systems. SIP also can be used in conjunction with other protocols such as Session Description Protocol “SDP” and Real-Time Protocol “RTP” to provide communications services.

With the advent of Voice over Internet Protocol (“VoIP”), VoIP turret devices have moved from a fixed environment of dedicated voice communications trading resources to a more virtualized and remotely accessible trading environment across multiple sites. This virtual environment allows trader mobility (e.g., to roam across multiple trading floors), and disaster recovery planning and resources (e.g., remote access to their private wires) to be shared dynamically, when and where required, across a global corporate enterprise.

FIG. 1 shows an overview of a previous generation soft-turret product. This system makes use of an external hardware media bridge **102** and a hard-turret **104** for one soft-turret instance to operate. Hard-turret **104** typically includes two handsets, commonly referred to as a Left Handset “LHS” and

2

a Right Handset “RHS” and multiple speaker devices. This allows a trader to have many independent voice conversations simultaneously. A soft-turret **106** is typically connected to a turret switching system **112** through a network such as the Internet **114** and is accompanied by a single phone device for voice delivery. Thus, soft-turret users have a single handset only.

To accommodate for this, a typical soft-turret product makes use of the external media bridge **102** to bridge the LHS to RHS so that one channel is used to connect to the private wire network **110** and the other channel is used to connect a public-switched-telephone-network or PSTN phone **108** over a PSTN network **116**. This solution is costly, however, because it requires an external piece of hardware and requires a hard-turret to be dedicated to the remote trader.

Typical soft phones have other practical limitations. For example, a soft-phone runs on a PC and is typically connected to an organization’s private branch exchange (PBX) and makes use of VoIP to communicate with the PBX. In such configurations, the soft-phone can exchange signaling (e.g., SIP) and media (e.g., RTP) with the PBX but does not have a dedicated data channel with the PBX for rendering a display to the user. Soft phones are typically also not capable of delivering multiple conversations (such as multiple handsets and speakers as is the case in turrets) simultaneously.

Another factor that must be taken into consideration when dealing with trader communications is voice quality. Voice quality requirements (such as voice cut-through times) in a trading environment are significantly more stringent than in an enterprise PBX environment. For instance, as soon as a trader seizes a private wire, voice communications must be established immediately (e.g., <50 ms) such that a single spoken syllable from the trader is not dropped.

Another bottleneck in existing trader systems is their computing systems. A trader’s personal computer may be running many voice applications simultaneously such as Voice and Instant Messaging clients, Web browsers with voice enabled Web pages, etc. In addition, the personal computer’s operating system is usually non-real time, which means it is not capable of switching tasks in real time in order to preempt low priority tasks to allow applications delivering voice media to run. Sending the voice to trader’s PSTN (Public Switched Telephone Network) phone line thus presents contention issues.

BRIEF DESCRIPTION

The example embodiments described herein meet the above-identified needs by providing methods, systems and computer-readable medium for providing a software-based trading turret.

In one example embodiment, a turret switching system communicates to a Web server, a turret device, and to a remote communications device via a first communications network. The Web server communicates to a client device via a second communications network. The client device controls switching across a plurality of lines.

In another example embodiment, a computer-readable medium having stored thereon sequences of instructions is provided. The sequences of instructions include instructions which when executed by a computer system causes the computer system to communicate to a Web server, a turret device, and to a remote communications device via a first communications network, communicate to a client device via a second communications network, and control switching across a plurality of lines.

US 8,189,566 B2

3

In yet another example embodiment, a method for providing a software based trading turret is provided by communicating, by a turret switching device, to a Web server, a turret device, and a remote communications device, via a first communications network. The method also provides communicating, by the turret switching device, to a client device via a second communications network; and controlling switching across a plurality of lines, by the turret switching device.

Further features and advantages, as well as the structure and operation, of various example embodiments of the present invention are described in detail below with reference to the accompanying drawings.

BRIEF DESCRIPTION OF THE DRAWINGS

FIG. 1 depicts a diagram of a prior art soft-turret system.

FIG. 2 depicts a system for delivering voice and data over separate communications channels in accordance with an embodiment of the present invention.

FIG. 3 depicts a system for delivering voice and data over a single data network, where the voice is delivered over a VoIP network using SIP and RTP protocols, in accordance with an embodiment of the present invention.

FIG. 4 depicts a system for delivering voice and data over a single data network, where a SIP phone is embedded inside soft-turret application, in accordance with an embodiment of the present invention.

FIG. 5 depicts a system for delivering voice and data using only one signaling protocol for employing voice and data sessions, in accordance with an embodiment of the present invention.

FIG. 6 shows the internal software architecture of a soft-turret Web server in accordance with an embodiment of the present invention.

FIG. 7 depicts an exemplary multi-tasking architecture of a soft-turret Web server in accordance with an embodiment of the present invention.

FIG. 8 illustrates an exemplary mechanism of button-press stimulus delivery from a soft-turret client to a server, in accordance with an embodiment of the present invention.

FIG. 9 illustrates an exemplary mechanism of event delivery from a soft-turret server to a client, in accordance with an embodiment of the present invention.

FIG. 10 depicts a screenshot of an exemplary soft-turret launched from within a Web browser on a PC, in accordance with an embodiment of the present invention.

The features and advantages of the example embodiments of the invention presented herein will become more apparent from the detailed description set forth below when taken in conjunction with the drawings in which like reference numbers indicate identical or functionally similar elements.

DETAILED DESCRIPTION

The example embodiments of the invention presented herein are directed to methods, systems and computer-readable medium for a software based trading turret, which are now described herein in terms of an example trading environment and a remote client. This is for convenience only and is not intended to limit the application of the present invention. In fact, after reading the following description, it will be apparent to one skilled in the relevant art(s) how to implement the following invention in alternative embodiments, such as security/law enforcement, utilities, healthcare, and customer support (e.g., contact centers) environments.

Generally, trader mobility and disaster recovery planning are provided by a system having a trading turret modeled as

4

two parts. One part renders a turret's Graphical User Interface (GUI) on a PC screen and accepts user stimuli while the other part delivers voice communications to the trader.

A soft-turret application is used to facilitate synchronization of both parts by communicating signaling information to/from a turret switching system over a data network such as the public Internet while voice communications are in progress over the PSTN.

FIG. 2 depicts an arrangement in which voice and signaling are communicated between a turret switching system and a remote client at a backup trading floor or trader's home. Particularly, voice is delivered between a turret switching system **214** connected to a private wire network **110** and a PSTN phone **209** through a PSTN **202**. Data communications that carry signaling such as display-refreshes and button selections (i.e., "presses") are delivered to/from a PC **208** over an IP network **204** such as the Internet.

A soft-turret application **206** running on PC **208** uses a secure form of HTTP (Hyper-Text Transfer Protocol) for exchanging data with a farm of Web servers **210** deployed in a trading floor backroom **212** environment. Soft-turret application **206** can be launched from a Web browser or as a standalone software application. Once launched, soft-turret application **206** renders a turret on the display of the PC **208**, such as one in the trading floor backroom **212**, i.e. hard-turret **218**.

Once soft-turret application **206** has been subscribed to Web server **210**, Web server **210** is enabled to deliver line status change notifications to the soft-turret application **206**. This allows soft-turret application **206** to listen for line state changes from the turret system **214** and ensures that the line states are consistent across both systems. For instance, a user of hard-turret **218** seizes a line located in the trading floor backroom, soft-turret application will receive a corresponding notification. For simplicity only one representative Web server is shown.

Several private lines can be coupled to turret switching network **110**. One or more of these lines can be fed to PSTN phone **209** through PSTN **202** allowing a trader to have multiple independent voice conversations simultaneously. Soft-turret application **206** is enabled to seize lines on the private wire network **110** and perform functions available on a hard-turret **218** from a remote PC **208**. Signaling information associated with the seized lines are communicated to/from the soft-turret application **206**.

For example, soft-turret application **206** can deliver button presses to Web server **210**, which in turn converts them into the communication protocol understood by the station cards installed in the turret switching system **214**. LHS/RHS bridging is performed inside media bridge **216**, eliminating the need for an external media bridge or a dedicated hard-turret.

By enabling turret switching system **214** to connect multiple lines on a private wire network **110** to a single PSTN phone **209**, a user can communicate to multiple parties that have been remotely connected by turret switching system **214** under the control of a remote client (e.g., PC **208**). This permits signaling information and voice connections to be synchronized.

FIG. 3 depicts a variation of the embodiment described above with respect to FIG. 1. In this arrangement, voice is delivered over a VoIP network using SIP and RTP protocols as opposed to over a PSTN. Particularly, turret switching system **314** delivers voice between a private wire network **110** and a SIP phone **309** via an IP network **304** (e.g., Internet).

Data communications that carry information from a PC **308**, such as display refreshes and button selections (i.e., "presses") also are delivered over IP network **304**. Particu-

US 8,189,566 B2

5

larly, a soft-turret application 306, running on a PC 308 uses a secure form of HTTP (Hyper-Text Transfer Protocol) for exchanging data with a farm of Web servers 310 deployed in the trading floor backroom 312. The soft-turret application 306 may be launched from either within a Web browser or as a standalone software application.

Once soft-turret application 306 has been subscribed to Web server 310, Web server 310 is enabled to deliver line status change notifications to the soft-turret 306. This allows soft-turret application 306 to listen for line state changes from the turret system 314 and ensures that the line states are consistent across both systems. For instance, a user of hard-turret 318 seizes a line located in the trading floor backroom, soft-turret application will receive a corresponding notification.

As described above, several private lines can be coupled to turret switching network 110. These lines can be fed to SIP phone 309 by the turret switching system 314 through IP network 304 allowing a trader to have multiple independent voice conversations simultaneously. In addition, soft-turret application 306 is enabled to seize lines on the private wire network 110 and perform functions available on a hard-turret 318 from a remote PC 308. Signaling information associated with the seized lines is communicated to/from the soft-turret application 306.

For example, soft-turret application 306 can deliver button presses to Web server 310, which in turn converts them into the communication protocol understood by the station cards installed in the turret switching system 314. LHS/RHS bridging is performed inside media bridge 316, eliminating the need for an external media bridge or a dedicated hard-turret.

By enabling turret switching system 314 to connect multiple lines on a private wire network 110 to a single SIP phone 309, a user can communicate to multiple parties that have been remotely connected by turret switching system 314 under the control of a remote client (e.g., PC 308). This configuration thus provides real-time synchronization of the voice and signaling data by separating each into two channels and communicating both channels over an IP network 404.

FIG. 4 depicts a system for delivering voice and data communications over a single network, IP network 404. In this embodiment, a SIP phone is embedded inside a soft-turret application 406 running, for example, on a PC 408. Soft-turret application 406 uses a SIP user agent ("UA") 407 to initiate and answer calls. The display refreshes and button selections are delivered over HTTP.

Voice communications are delivered over a VoIP network using SIP and RTP protocols while the data communications such as display refreshes and button selections are delivered using a secure form of HTTP (e.g., HTTPS).

The soft-turret application 406 in this embodiment may be launched from either within a Web browser or as a standalone software application. The Web server 410 delivers line status change notifications to the soft-turret 406, and the soft-turret 406 delivers button presses to the Web server 410. Web server 410, in turn, converts user button press commands into a proprietary communication protocol understood by the station cards installed in the turret switching system 414. In this example, instead of using a PSTN or SIP hard-phone, a trader uses a microphone and speaker(s) connected to PC 408 to exchange voice with the private wire network 110.

Once soft-turret application 406 has been subscribed to Web server 410, Web server 410 is enabled to deliver line status change notifications to the soft-turret application 406. This allows soft-turret application 406 to listen for line state changes from the turret system 414 and ensures that the line states are consistent across both systems. For instance, a user

6

of hard-turret 418 seizes a line located in the trading floor backroom, soft-turret application will receive a corresponding notification. For simplicity only one representative Web server is shown.

Several private lines can be coupled to turret switching network 110. One or more of these lines can be fed to SIP UA 407 through IP network 404 allowing a trader to have multiple independent voice conversations simultaneously. Soft-turret 406 is enabled to seize lines on the private wire network 110 and perform functions available on a hard-turret 418 from a remote PC 408. Signaling information associated with the seized lines are communicated to/from the soft-turret application 406.

For example, soft-turret application 406 can deliver button presses to Web server 410, which in turn converts them into the communication protocol understood by the station cards installed in the turret switching system 414. LHS/RHS bridging is performed inside media bridge 416, eliminating the need for an external media bridge or a dedicated hard-turret.

By enabling turret switching system 414 to connect multiple lines on a private wire network 110 to a single SIP UA 407, a user can communicate to multiple parties that have been remotely connected by turret switching system 414 under the control of a remote client (e.g., PC 408). This permits signaling information and voice connections to be synchronized.

FIG. 5 depicts a system for delivering voice and data using only one signaling protocol for employing voice and data sessions. By employing one signaling protocol for voice and data sessions a converged client is provided. Particularly, no separate data channel (e.g., HTTPS) is required; only SIP/RTP communications are used. Particularly, line status notifications and button presses are delivered as SIP messages and line status notifications and button press requests that were previously delivered over HTTPS are now also delivered over SIP.

Referring still to FIG. 5, a system for delivering voice and data over a single data network, where a SIP phone is embedded inside soft-turret application, is shown. In this embodiment, a trader uses a PC 508 running a soft-turret application 506 for both voice communications and data. Soft-turret application 506 includes a SIP UA 507 to initiate and answer calls between peers. Voice is delivered from PC 508 to a turret switching system 514 over IP network 504 using SIP and RTP protocols. The soft-turret application 506 may be launched from either within a Web browser or as a standalone software application.

Turret switching system 514 delivers line status change notifications to the soft-turret 506, and the soft-turret 506 delivers button presses to the turret switching system 514. In this embodiment, user button press commands are constructed such that they are understood by the station cards installed in the turret switching system 514. A trader use a microphone and speakers connected to the PC 508 to exchange voice.

As illustrated above, none of the embodiments require an external media bridge or a dedicated hard-turret. Instead, LHS/RHS bridging is now done inside a DSP based media bridge placed on the station card through which turret stations connect to the switching fabric. In addition, once a trader is logged in, a communication channel to the PSTN phone, SIP phone, UA, as the case may be, is established. Whether and which line (e.g., from private wire network 110) is communicated to the soft-turret is controlled by the soft-turret application.

FIG. 6 depicts the internal software architecture of an example soft-turret Web server 600 in more detail. As shown

US 8,189,566 B2

7

in FIG. 6, the architecture employs an HTTP server **606** for terminating the HTTP protocol and for serving Web pages. Soft-turret Web server **600** also includes a soft-turret proxy **604** which collects messages communicated over the HTTP protocol and, in turn, communicates them to one or more services **602**. A Who's Who component **610** allows a trader to login using, for example, an alphanumeric AoR (Address of Record) as opposed to a numeric TRID (Trader ID). The Who's Who component **610** may also be in communication with a customer authentication server **612** which provides authentication services. This allows soft-turret Web server **600** to utilize customer authentication service **612** using, for example, the Kerberos protocol to facilitate trader log-on. The Kerberos protocol can also be used to encrypt communications between the soft-turret Web server and the turret switching system.

As described above with respect to FIGS. 2-5, the same functionality previously performed on a hard turret is now performed on a soft-turret. When stimulus from a soft-turret is received by HTTP server **606** via a network (e.g., the public Internet) the stimulus is communicated to soft-turret proxy **604**. Soft-turret proxy **604**, in turn, directs this signaling to either a service **602**, a licensing processor **608** or a Who's Who processor **610**.

Services provide signaling and other information such as authentication information to the turret switching system (e.g., **214**, **314**, **414**, **514**).

In addition, soft-turret Web server **600** can include a licensing component **608** to facilitate authentication. For example licensing component **608** can limit the number of users that are able to login into and use the system, for example based on a predetermined number of allowed users.

The soft-turret Web server also includes several application tasks or processes modules. A single soft-turret web server can support hundreds of simultaneous client connections and each client connection is supported through a set of processes that are run on the operating system.

FIG. 7 illustrates an example flow of various tasks involved in soft-turret Web server operations. As shown in FIG. 7, signaling can be received from a soft-turret client via a soft-turret proxy **714**. In turn, soft-turret proxy **714** communicates the signaling to one or more services, such as an applications worker **704**, telephony worker **708** and persistence worker **712**. Particularly, the group leader of each worker (i.e., an applications group leader **702**, a telephony group leader **706** and a persistence group leader, respectively) create their respective workers. Signaling information from the workers, in turn is communicated back to the soft-turret application through soft-turret proxy **714**.

FIG. 8 illustrates an exemplary mechanism of button-press stimulus delivery from a soft-turret client to a server. In this example, a user of a soft-turret client **802** selects a button, such as a button to answer a line. The button data associated with the selection is communicated from the PC to an HTTP server **804**, which in turn communicates the data to a soft-turret proxy **806**. Soft-turret proxy **806** forwards the data to one or more services. A service, such as an application worker **808**, communicates the data to a turret switching system using an appropriate signaling protocol.

FIG. 9 illustrates an exemplary mechanism of event delivery from a soft-turret server to a client. In this example, signaling information about an incoming call is communicated to an application worker **902**. A soft-turret client **908** polls an HTTP server **906** for line status change events. HTTP server, in turn, forwards the event poll to a soft-turret proxy **904**. The HTTP poll requests are pending at the server until a line state change occurs. As soon as a line state change occurs,

8

the server returns a response to the client, therefore, providing a real-time state change delivery mechanism. This response contains the specific information about line state changes. This technique is commonly known as HTTP long-poll or "Comet".

Events are communicated by the application worker **902** to the soft-turret proxy **904**, which, in turn, communicates an event response to an HTTP server **906**. The events are accepted and queued within the HTTP server so they can be dispatched to clients in batches. HTTP server **906** communicates the event response to a soft-turret client **908**.

FIG. 10 depicts a screenshot of an exemplary soft-turret launched from within a Web browser on a PC, in accordance with an embodiment of the present invention. As shown in FIG. 10, various sets of buttons and indicators are provided (**902**, **904**, **906**, **908**, **910**). These buttons and indicators provide the same features found on a hard turret, such as turrets **218**, **318**, **418** and **518** described above with respect to FIGS. 2-5, which are typically found on a trading floor. For example, a dialpad button **902** provides a user with a set of numbers used to make a call, and multiple lines can be selected through buttons **904**. The same options available on the physical hard turret also are available on the GUI of the soft-turret through the options set of buttons **906**. Similarly, transferring, conferencing, hold, release and mute button features are signaled through the GUI using button sets **908** and **910**, respectively.

Synchronization of the voice and signaling data is achieved by seizing a voice line between a PSTN phone, a SIP phone or a SIP UA, as the case may be and communicating control and status data on a separate channel. Multiple lines, for example on a private wire network, may be bridged by a turret switching system. Similarly, the LHS and RHS may be bridged through a media bridge within the turret switching system.

A conference including of these voice lines may be provided on the seized line between the turret switching system and the remote phone, with the control and status (i.e., signaling) being communicated through a separate channel between a soft-turret application and a web server in communication with the turret switching system, using a GUI such as the one shown in FIG. 10.

The example embodiments of the invention may be implemented using hardware, software or a combination thereof and may be implemented in one or more computer systems or other processing systems. However, the manipulations performed by these example embodiments were often referred to in terms, such as entering, which are commonly associated with mental operations performed by a human operator. No such capability of a human operator is necessary, in any of the operations described herein. Rather, the operations may be completely implemented with machine operations. Useful machines for performing the operation of the example embodiments presented herein include general purpose digital computers or similar devices.

From a hardware standpoint, a CPU typically includes one or more components, such as one or more microprocessors, for performing the arithmetic and/or logical operations required for program execution, and storage media, such as one or more disk drives or memory cards (e.g., flash memory) for program and data storage, and a random access memory, for temporary data and program instruction storage. From a software standpoint, a CPU typically includes software resident on a storage media (e.g., a disk drive or memory card), which, when executed, directs the CPU in performing transmission and reception functions. The CPU software may run on an operating system stored on the storage media, such as, for example, UNIX or Windows (e.g., NT, XP, Vista), Linux, and the like, and can adhere to various protocols such as the

US 8,189,566 B2

9

Ethernet, ATM, TCP/IP protocols and/or other connection or connectionless protocols. As is well known in the art, CPUs can run different operating systems, and can contain different types of software, each type devoted to a different function, such as handling and managing data/information from a particular source, or transforming data/information from one format into another format. It should thus be clear that the embodiments described herein are not to be construed as being limited for use with any particular type of server computer, and that any other suitable type of device for facilitating the exchange and storage of information may be employed instead.

A CPU may be a single CPU, or may include plural separate CPUs, wherein each is dedicated to a separate application, such as, for example, a data application, a voice application, and a video application. Software embodiments of the example embodiments presented herein may be provided as a computer program product, or software, that may include an article of manufacture on a machine accessible or computer-readable medium (i.e., also referred to as "machine readable medium") having instructions. The instructions on the machine accessible or machine readable medium may be used to program a computer system or other electronic device. The machine-readable medium may include, but is not limited to, floppy diskettes, optical disks, CD-ROMs, and magneto-optical disks or other type of media/machine-readable medium suitable for storing or transmitting electronic instructions. The techniques described herein are not limited to any particular software configuration. They may find applicability in any computing or processing environment. The terms "machine accessible medium", "machine readable medium" and "computer-readable medium" used herein shall include any medium that is capable of storing, encoding, or transmitting a sequence of instructions for execution by the machine (e.g., a CPU or other type of processing device) and that cause the machine to perform any one of the methods described herein. Furthermore, it is common in the art to speak of software, in one form or another (e.g., program, procedure, process, application, module, unit, logic, and so on) as taking an action or causing a result. Such expressions are merely a shorthand way of stating that the execution of the software by a processing system causes the processor to perform an action to produce a result.

While various example embodiments of the present invention have been described above, it should be understood that they have been presented by way of example, and not limitation. It will be apparent to persons skilled in the relevant art(s) that various changes in form and detail can be made therein. Thus, the present invention should not be limited by any of the above described example embodiments, but should be defined only in accordance with the following claims and their equivalents.

In addition, it should be understood that the figures are presented for example purposes only. The architecture of the example embodiments presented herein is sufficiently flexible and configurable, such that it may be utilized (and navigated) in ways other than that shown in the accompanying figures. It is also to be understood that the procedures recited in the claims need not be performed in the order presented.

What is claimed is:

1. A communications system, comprising:

a turret switching system constructed to communicate to a Web server, a turret device, and to a remote communications device via a first communications network, the Web server being constructed to communicate to a client

10

device via a second communications network, and the client device constructed to control switching across a plurality of lines; and

an interface having a button sheet corresponding to a plurality of line selectors and constructed to seize a corresponding line by causing the client device to communicate a predetermined message to the turret switching system over the second communications network.

2. The communications system according to claim 1, wherein the first communications network is a PSTN network and the remote communications device is a PSTN device.

3. The communications system according to claim 1, wherein the corresponding line is communicatively coupled to the communications device via the first communications network upon selection of one of the plurality of line selectors.

4. The communications system according to claim 1, wherein the predetermined message is communicated over the second communications network using SIP.

5. The communications system according to claim 1, wherein switching of the corresponding line is performed in the turret switching system.

6. The communications system according to claim 1, wherein the first communications network and the second communications network are IP networks.

7. A non-transitory computer-readable medium having stored thereon sequences of instructions, the sequences of instructions including instructions which when executed by a computer system causes the computer system to:

communicate to a Web server, a turret device, and to a remote communications device via a first communications network;

communicate to a client device via a second communications network;

control switching across a plurality of lines; and

seize a corresponding line by causing the client device to communicate a predetermined message to a turret switching system over the second communications network.

8. The non-transitory computer-readable medium of claim 7, wherein the first communications network is a PSTN network and the remote communications device is a PSTN device.

9. The non-transitory computer-readable medium of claim 7, wherein the corresponding line is communicatively coupled to the communications device via the first communications network upon selection of one of a plurality of line selectors.

10. The non-transitory computer-readable medium of claim 7, wherein the predetermined message is communicated over the second communications network using SIP.

11. The non-transitory computer-readable medium of claim 7, wherein switching of the corresponding line is performed in the turret switching system.

12. The non-transitory computer-readable medium of claim 7, wherein the first communications network and the second communications network are IP networks.

13. A method for providing a software based trading turret, comprising the steps of:

communicating, by a turret switching device, to a Web server, a turret device, and a remote communications device, via a first communications network;

communicating, by the turret switching device, to a client device via a second communications network;

controlling switching across a plurality of lines, by the turret switching device; and

US 8,189,566 B2

11

seizing a corresponding line by communicating a predetermined message to a turret switching system over the second communications network.

14. The method of claim **13**, wherein the first communications network is a PSTN network and the remote communications device is a PSTN device.

15. The method of claim **13**, wherein the corresponding line is communicatively coupled to the communications device via the first communications network upon selection of one of a plurality of line selectors.

12

16. The method of claim **13**, wherein the predetermined message is communicated over the second communications network using SIP.

17. The method of claim **13**, wherein switching of the corresponding line is performed in the turret switching system.

18. The method of claim **13**, wherein the first communications network and the second communications network are IP networks.

* * * * *

UNITED STATES PATENT AND TRADEMARK OFFICE
CERTIFICATE OF CORRECTION

PATENT NO. : 8,189,566 B2
APPLICATION NO. : 12/476490
DATED : May 29, 2012
INVENTOR(S) : Aseem Bakshi et al.

Page 1 of 2

It is certified that error appears in the above-identified patent and that said Letters Patent is hereby corrected as shown below:

In the Specification

COLUMN 1:

Line 64, “us” should read --use--.

COLUMN 2:

Line 62, “causes” should read --cause--.

COLUMN 4:

Line 44, “are” should read --is--.

COLUMN 5:

Line 12, “instance,” should read --instance, when--.

COLUMN 6:

Line 12, “are” should read --is--; and

Line 53, “use” should read --uses--.

COLUMN 7:

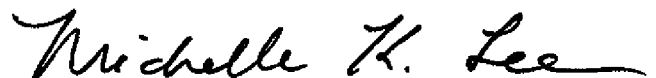
Line 34, “processes” should read --process--; and

Line 38, “and” should read --an--.

COLUMN 8:

Line 34, “of” should be deleted.

Signed and Sealed this
Fifteenth Day of April, 2014



Michelle K. Lee
Deputy Director of the United States Patent and Trademark Office

CERTIFICATE OF CORRECTION (continued)

Page 2 of 2

U.S. Pat. No. 8,189,566 B2

In the Claims

COLUMN 10:

Claim 7, Line 30, “causes” should read --cause--.

EXHIBIT

2

WebRTC in the Financial Sector – An Interview with Leo Papadopoulos from Cloud9 Technologies

June 22, 2015 / in [Markets & People](#), [VoIP](#) / [Add Comment](#)

WebRTC is perceived by many as a technology that allows for in-browser, zero download, voice and video communication. While this is true, it is also an open source media engine that can be used in different ways. Cloud9 Technologies uses WebRTC with a twist. This is because of 2 reasons: their application and the market in which they operate.

Leo Papadopoulos is CTO & Technology Evangelist for Cloud9 Technologies. We both know each other from our previous business life so I asked him to share the details of how they used WebRTC and other technologies for building their service.



Q. What is Cloud9 Technologies all about?

We see ourselves as pioneers in delivering next generation, high performance Cloud Communications Services for targeted segments of the B2B market. We specifically focus on segments that need reliability, security, compliance, and superior support. Our early adopters are financial institutions and the voice and messaging services used by institutional

traders, called trader voice. The Cloud9 (c9tec) solution lets firms transition from expensive, propriety hardware and private networks to applications deployed on commercially available devices. We deliver this as a service with an easy to understand and deploy SaaS business model.

Q. How do you make use of WebRTC in your service, what was your main reason to select this technology?

We wanted to focus on our application and we needed a voice engine. We looked at traditional stacks like SIP but they were way too stringent, required expensive and inflexible application servers, forced us to use signaling that simply did not meet our needs, and did not include the actual voice engine itself. We quickly realized that all the existing protocols were really just variants of old telephony protocols. Quite simply, you cannot develop modern communications services using protocols made to simulate phone lines.

We looked at WebRTC. It allowed voice & video connections, it included methods for ICE/STUN/TURN, it included SRTP for secure voice, it included OPUS which is a modern voice CODEC, and importantly it did *not* force us to use a predefined signaling protocol. It was almost as if someone designed this just for us.

We compiled WebRTC into our application much like the browser manufacturers do. However, it is on top of this that we added our intellectual property that allows us to deliver the unique and special needs of our customers. We layered in a user interface specific to our customers. Reliability and automatic recovery to keep communications continuous. Instrumentation that allows us to deliver proactive support, along with compliance features that regulatory agencies require of our customers, and many more features.



Q. Your use of WebRTC is not common, you have decided not to benefit from the zero download option and compile it into your application. What are the reasons for this decision and do you see this changing in the future?

Most of the attention that WebRTC gets comes from companies that provide toolkits and APIs and also obviously a lot of focus on call centers. We would characterize these opportunities as consumer oriented or B2C at best. That environment has a higher tolerance for browser based solutions and modest support levels and reliability.

Our business focus is the enterprise and B2B services. So when we looked at WebRTC, we said this is cool but how can we make it work at an enterprise level? We've done that and it is one of the things that distinguishes Cloud9.

As for the zero download option – our view is that browsers are disposable and

not suitable for an enterprise grade B2B service. They are a great option for remote access but if you really want to deliver a service to a high end clientele it needs to be packaged in an application with the performance, service levels and support expected by Fortune 100 companies. This is particularly true for our target markets.

Q. What are you using for signaling on top of WebRTC?

Just like a lot of the WebRTC community we use a WebSocket server for signaling. We implement this using a Node.js server which has amazing performance. The actual signaling that we use is proprietary and is part of what makes our application special. It is within this signaling that we take some historically very complex features and implement them in a new much simpler method. This would have been impossible to do using traditional VoIP signaling protocols. However, the combination of WebSockets and Node.js unleash amazing new power.

Q. What technologies are you using in your backend?

There are many technologies available to developers today. The only thing holding anyone back is imagination. Putting these tools together in the right way and adding your own 'special sauce' will lead to success. Here are just some of the tools in our toolbox:

- Voice/Video Engine – WebRTC, OPUS, VP9
- Web services and signaling – HTML5, WebSocket, Apache, Node.js
- Development – C#, .NET, jQuery, PHP, MySQL, C++, Visual Studio
- Amazon Web Services – EC2, VPC, S3, RDS, Route 53, CloudWatch, SNS, IAM, & more.

We think this is pretty cool. You can build and deploy a high performance global communication solution without ever talking to a carrier or telecommunications equipment provider.

You could not do that even 10 years ago. In today's world if you are building your

own infrastructure, then you're probably not investing enough on your service or application and that means that someone else will probably beat you in the market. The tools are there and we all must use them.

Q. Are you using any 3rd party services behind the scenes?

We are using:

- Amazon Web Services for all our servers, storage, and databases.
- Twilio for global SMS messaging.

We are also investigating Microsoft Azure because they seem to have some really nice new products that allow for much better solutions for secure encrypted storage in the cloud.

Q. What would you want to see coming into WebRTC as roadmap?

I think for us the most important thing for WebRTC is voice quality and reliability. To this end here are a few specific features we would like to see:

- 5msec packet support, for codecs that support it, including 5msec audio frame support for AEC, noise reduction, etc. More than the CODEC all the audio algorithms should support small packets for latency critical environments.
- RTCP-XR VoIP metric and statistics summary block report support. This allows us to better deliver superior support to our customers by seeing their problems before they do.
- Customizable audio filters, for example a multiband EQ.
- Configurable STUN ping period. The default period is too fast and uses too much bandwidth and CPU as a result. This has been **addressed at the last IETF meeting** but hasn't been finalized and guidelines have not been published yet.

Q. Any future plans for Cloud9 Technologies you would

want to share?

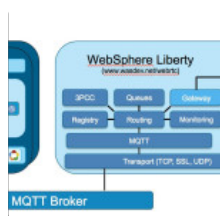
We see a very broad opportunity for WebRTC. In fact we think it's transformational and will ultimately change the way you think about communications and messaging and everything from browsers and call centers to the very way you make a call or message with your cell phone.

For Cloud9:

- Our focus is on the high end of the B2B market where service levels and performance matter.
- Our services are designed for work groups and teams, as well as individuals. Team engagement and collaboration is a key part of any enterprise. Technologies like WebRTC let companies like Cloud9 deliver secure and compliant services in today's mobile and global environment.

We support team collaboration both within and between enterprises. Much attention has been placed on collaboration within an enterprise but many of these services miss the mark when it comes to engagement between companies. We think this is a largely underserved market opportunity and look to leverage WebRTC and Cloud services to transform the way business counterparties connect and communicate.

You may also like:



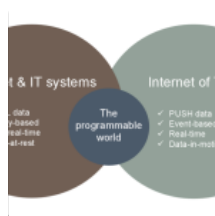
IBM WebSphere and WebRTC – An Interview with Brian Pulito from IBM



WebRTC Based UC for Service Providers – An Interview with Elena Carstoiu from 4PSA



OpenWebRTC: An Interview with Stefan Alund



An Interview with Veselin Pizurica from waylay.io

Leave a Reply

Your email address will not be published. Required fields are marked *

Comment

Name *

Email *

Website

POST COMMENT

☐ Notify me of follow-up comments by email.

☐ Notify me of new posts by email.

Search this website ...

SUBSCRIBE BY EMAIL

SUBSCRIBE TO RSS

ABOUT ME



Amir Zmora

Blogging about new technology trends
and their impact on markets and people.

[read more](#)

Follow [@AmirZmora](#)

My Other Websites

WebRTC Standards

WEBRTC**INDEX**

Categories

[IoT](#)[Markets & People](#)[Mobile](#)[VoIP](#)[WebRTC Standards](#)

READ MY POSTS ON OTHER BLOGS

[NoJitter](#)[BlogGeek.me Posted & Mentioned](#)[UC Strategies](#)[AudioCodes Blog](#)[Upperside Conferences](#)[WebRTC World](#)

© 2014 THE NEW DIAL TONE

Designed by [Katika](#)

EXHIBIT

3

Trade Secrets Policy

An important part of the success and value of IPC's business is contained in its confidential business information and trade secrets. It is very important that all employees take appropriate care to protect such confidential business information and trade secrets to ensure that:

- they do not end up in the possession of third parties that should not have access to them, such as competitors
- they do not become public knowledge and therefore lose the protection granted by applicable laws
- all uses of confidential information and trade secrets are necessary and appropriate for the business interests of IPC

For the purposes of this Policy references to employees shall include consultants and contractors.

IPC takes the protection of its confidential business information and trade secrets very seriously. In addition to any obligations imposed by applicable law, IPC requires all employees to agree to confidentiality and non-disclosure obligations, whether contained in an employment agreement or a separate agreement. In addition to these obligations, employees are required to comply with the spirit and the letter of this Policy.

IPC employees who fail to comply with this policy may be subject to disciplinary procedures, up to and including termination of employment from IPC. In addition, IPC will pursue any and all remedies against the employee available to it under applicable laws where there has been misuse of trade secrets or confidential information.

Any breach of this Policy or any other obligations, policies and/or procedures relating to trade secrets and/or confidential information, may cause substantial and irreparable harm to IPC. To the extent permitted by applicable law, IPC may seek injunctive relief and any other relief that it deems necessary and appropriate under the circumstances to protect its rights.

What Constitutes Trade Secrets and Confidential Business Information

In the US under federal and state law, a trade secret is generally any formula, pattern, device or compilation of information that (i) is used in a business, (ii) is unknown to others outside that business, (iii) gives the business a competitive advantage, and (iv) is the subject of reasonable efforts to maintain its secrecy. A trade secret does not have to be unique or novel; it need only provide a distinct economic value to IPC. Confidential business information includes trade secrets as well as other proprietary information that may not rise to the level of a trade secret. Trade secrets and confidential business information may be defined differently depending on the jurisdiction and IPC will always seek the maximum protection of its business afforded by the local courts.

The following is a non-exhaustive list of items that qualify as IPC trade secrets and/or confidential business information:

- IPC customer lists
- The names and qualifications of IPC's development partners and/or contractors

- Any aspect or description of IPC's relationships with its customers, for example, product or service purchases, pricing policies, price lists, discounts, orders and revenue
- Computer systems, software (object code and source code) and databases
- Internal specifications, technical processes, testing procedures, diagrams, designs, drawings, models, and any other techniques, developments, improvements, inventions, and processes that are, or may be, produced in the course of IPC's engineering or other operations
- Marketing development and research plans
- Manufacturing processes or techniques
- Financial, accounting, and legal information
- Any other information not generally and publicly known regarding IPC or its operations, products, suppliers, markets, sales, costs, profits or customers, or other information acquired, disclosed or made known to any employees or agents during the course of their employment or agency that, if used or disclosed, could adversely affect IPC's business or give its competitors a commercial or economic advantage.

Information will not be regarded as trade secrets and/or confidential information if such information:

- is available in the public domain (provided this has not happened because of a breach of any obligations regarding confidentiality or trade secrets)
- was independently created by a third party
- is trivial or obvious

Treatment of Trade Secrets and Confidential Information

Employees owe IPC a high duty of loyalty. Employees must keep trade secrets and confidential business information confidential both during and after termination of employment with IPC for whatever reason (provided that in the case of confidential business information it shall be kept confidential for the maximum period permitted by applicable laws).

To assist in your continued compliance with your confidentiality obligations, here are a few tips:

- Protect your computer password; do not share it with anyone outside IPC or anyone within IPC other than as specifically directed by the IT Department.
- Keep sensitive files secure when you are away from your office or desk; lock your office door when you leave at the end of the day.
- Do not share confidential business information (including e-mails) or trade secrets with anyone outside of IPC unless there is a legitimate business reason for them to know the information and, only then, if they relate in any way to trade secrets with the permission of your supervisor or the Legal Department. This restriction includes friends, acquaintances and former IPC employees.
- Be careful (i.e., thorough) when destroying/discarding documents containing confidential business information and/or trade secrets.

- Identify as "Confidential and Proprietary Information of IPC" in boldfaced letters all documents created and/or distributed by you which contain confidential business information or trade secrets.
- Be vigilant in guarding your Blackberry, laptop and other portable devices which contain sensitive IPC information.
- Do not share IPC confidential business information with third parties (including consultants) unless they have signed a standard IPC non-disclosure agreement (available on the intranet site).
- Do not share IPC trade secrets with third parties (including consultants) unless they have signed a standard IPC non-disclosure agreement (available on the intranet site) and you have permission from the Legal Department.
- Do not engage in sensitive business conversations in elevators or other public spaces where third parties may eavesdrop. We are in a narrow, discreet market -- you never know who is listening.

This is not an exhaustive list; we encourage each of you to implement additional protective steps that can be taken within your particular area.

Employees are not authorized to access or use IPC's trade secrets and/or confidential business information for any purpose other than those specifically authorized by IPC. In no circumstances may trade secrets or confidential business information be accessed or used for the employee's own benefit, for competitive purposes or for non-IPC business purposes. Accessing IPC's trade secrets and/or confidential business information with the intent of using them in a manner that is in breach of the employee's duties or obligations owed to IPC is also prohibited.

Employees shall not discuss any matter that constitutes a trade secret with anyone other than an employee of IPC who has a need to know such information without the prior written consent of the IPC General Counsel. An employee shall not discuss any matter that is of a confidential nature with anyone other than an employee of IPC or a third party (where such third party has been specifically authorized by IPC and is under a duty to protection such confidential information) who has a need to know such information. Any discussions that do take place must be appropriate in both their content and location. Employees should never discuss any confidential business information or trade secrets in a public place. Without limiting the scope of the previous sentences confidential business information and trade secrets should never be discussed with a competitor of IPC (including its employees, contractors and agents). Such discussions may cause IPC and the employee personally to have violated applicable Antitrust/Competition laws.

Each employee has a responsibility to take reasonable precautions regarding the security and safekeeping of confidential business information and trade secrets, including but not limited to marking all plans, internal PowerPoint presentations and other confidential materials "confidential" and locking up or otherwise securing such information. Employees are responsible for the security of any passwords or encryption keys they are given or create to protect any IPC confidential information or trade secrets. You should not share or allow others to use such passwords or encryption keys.

Employees shall not disclose or share confidential business information and/or trade secrets through social media such as Facebook, LinkedIn, Twitter etc. Please also refer to IPC's Social Media Policy. Employees should think carefully about the media through which they communicate confidential information and/or trade secrets. Where practicable, avoid sending unprotected electronic versions of confidential business information. Electronic versions of trade secrets must never be sent without appropriate encryption or protection.

Every employee must disclose fully to IPC any inventions, ideas, works of authorship, and other trade secrets that are made, developed, and/or conceived by him or her alone or jointly with others or that arise out of or relate to his or her employment or other relationship with IPC. All such inventions, ideas, works of authorship, copyrights, and other trade secrets are and shall remain the sole property of IPC. Each employee must agree to execute and deliver to IPC such assignments, documents, agreements, or instruments that IPC may require from time to time to evidence its ownership of the results and proceeds of the Employee's services and creations.

Upon termination of employment or at IPC's request, each employee shall promptly return to IPC all memoranda, notes, records, reports, technical manuals, and any other documents (and all copies thereof) in his or her possession, custody, or control relating to IPC's business including IPC trade secrets. IPC may require the employee to certify that they have complied with this obligation.

Further Information/Advice

Please contact the Legal Department with any questions, comments or concerns regarding this Policy or the use of confidential information or trade secrets.

EXHIBIT

4



Go to...

Cloud9 Introduces Gateway Service to Help Firms Connect with More Counterparties and Migrate Away from Turrets

Posted by Steve Kammerer on May 2, 2016 6:17:54 PM

Tweet

Share

45



Cloud9 is now introducing Gateway: a service enabling traders to toss their turrets but still stay connected with counterparties, even if they are not yet using Cloud9.

The Cloud9 Gateway service provides system interoperability between existing trader voice circuits and the Cloud9 community by converting traditional TDM T1/E1 to WebRTC protocol. Cloud9 users can seamlessly connect with trading firms not yet established on Cloud9's service and maintain their functionality for both shout-downs and ring downs.

The Gateway Service supports two deployment models: **hosted** or **on-premises**. Both configurations offer various features and benefits, although Cloud9 recommends reviewing your technical and support needs to determine which option is best for your firm.

Advantages of the Cloud9 Gateway service include:

- Ability to expand your trading network and connect to counterparties on and off Cloud9.
- Rapid migration from trading turrets onto the Cloud9 service
- Improved disaster recovery capabilities by giving firms the ability to maintain connectivity during disaster scenarios.

Hosted Service

The hosted gateway option allows Cloud9 users to access traditional trader voice circuits by connecting their circuits to Cloud9 at a hosted facility. Pricing for this service is on a per month basis.

Benefits of a Hosted Service Include:

- Best support model for firms looking to relocate offices while transitioning to Cloud9.
- Circuits can be accessed by Cloud9 users within the firm from any location once they are connected to carriers.
- Ease of management by deploying inside a controlled data center environment vs on-premise.
- Loss of Internet or catastrophic event has no effect on operations due to redundancy within an off-site facility.
- Upgrades, patches and new features are included.

On-Premise Service

The on-premise service option is for customers that prefer not to move their existing trader voice circuits to a hosted data center. It also requires that a suitable IT environment be established for installing the Cloud9 gateway. This service option also gives customers a choice to purchase the service through monthly recurring fees, or as a single one-time charge.

Benefits for The On-Premise Service Include:

- No requirement to coordinate circuit relocation with telecom carriers - lines can be managed on site.
- Available for Cloud9 customers regardless of location.
- Upgrades, patches and new features are included.

To learn more about Cloud9, check out this quick introduction.

Topics: Cloud9 Product, Cloud9 News

First Name*

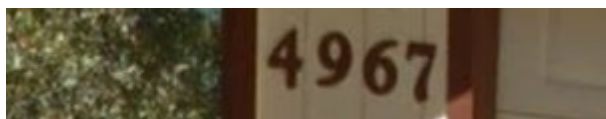
Last Name

Email*

Website

Comment*

☐ Subscribe to follow-up comments for this post



Type the text

[Privacy & Terms](#)



SUBMIT COMMENT

The Next Generation of Trader Voice Is Here

It's Time for Cloud9

Cloud9 is transforming the trading floor with modern technology to power trader voice capabilities and link financial communities together from all over the world.

Join 2000 traders, across 350 firms in 21 countries on Cloud9 to:

- **Reduce trader voice** costs by up to 50%.
- **Tap into a powerful global directory** to access new trading parties via.
- **Improve trader call quality** with HD audio
- **Achieve higher levels of security** with triple encryption technology.
- **Eliminate redundant, costly** back up equipment with a service that works anywhere the internet is available.

Subscribe to Email Updates

Email*

SUBSCRIBE

Recent Posts

- Jerry Starr Profiled in The Native Society
- Vote for Cloud9 in the 2016 Waters Rankings!
- Cloud9 CTO Discusses Impact of WebRTC on the Future of Communications
- Cloud9 Profiled in WatersTechnology for Sell-Side Tech Award
- Terry Clontz Joins Cloud9's Board of Directors
- Cloud9 Introduces Gateway Service to Help Firms Connect with More Counterparties and Migrate Away from Turrets
- Introducing Enhanced Encryption and Cloud9 Playback Features
- AlleyWatch Interviews Cloud9 President Greg Kenepp
- Cloud9 Wins Best Sell-Side Trading Communication System
- Customer Appreciation Happy Hour in Houston

Posts by Topic

- Cloud9 News (16)
- Cloud9 Awards (3)
- Cloud9 Product (3)
- Cloud9 Events (2)
- Cloud9 Technology (2)

[see all](#)

Cloud communication services for businesses and business communities.

CUSTOMERS

Help

Contact Us

LEGAL

Terms of Service

SLA

Privacy Policy

COMPANY

About Us

Open Source

Copyright 2016 Cloud9 Technologies LLC | All Rights Reserved

